Novel Trilateral Approach for Depth Map Spatial Filtering

Alexander Voronov
Moscow State University
Graphics & Media Lab
avoronov@graphics.cs.msu.ru

Dmitriy Vatolin
Moscow State University
Graphics & Media Lab
dmitriy@graphics.cs.msu.ru

Maxim Smirnov
YUVsoft Corp.
ms@yuvsoft.com

ABSTRACT

In this paper, we present our approach for spatial filtering of depth map extracted from camera motion. An original depth map may have some artifacts owing to imperfect motion estimation. Our goal was to make the depth map uniform in smooth areas and to refine object boundaries without blurring edges. To solve this problem we propose the trilateral filter, whose convolution kernel is composed of a distance kernel, a color-based kernel and a depth-based kernel. Experiments demonstrate that this approach yields rather good results. Also, we compare our results with those of a typical bilateral filter.

Keywords

depth map, disparity map, trilateral filtering, spatial filtering, post-processing, 3D video

1 INTRODUCTION

One of the widely used methods of creating 3D video involves changing the image parallax using a depth map. This process requires information regarding the distance between the camera and the objects in the scene. A depth map is a visualization of that distance for every pixel in the image: more-distant spots are represented using a darker color. Generally, the problem of creating of depth map from a single image is insoluble. So, until recently, depth maps were painted manually by stereo artists and composers in most cases – a task that required much time and money. But in some cases, depth maps can be created on the basis of information from a scene. Some such approaches apply machine learning algorithms to extract the information from a rather large set of images in different scales [Sax06]. Also, [Zhu09] proposes an approach that uses the fact that the camera is typically focused on foreground objects, so that objects have sharp edges: with increasing distance from the camera, object boundaries become blurrier. Another approach is to restore depth using the geometric properties of a scene: for example, by taking into account the vanishing point, horizon line, vertical lines and so on. This technique is presented in [Bat04] and [Jun10]. For scenes with camera motion we can create a depth map by applying an optical flow algorithm and analyzing how objects are moving in a scene. For example, if the camera is panning, an object's displacement in a given frame relative to the previous frame depends on that object's distance from the camera. This approach is described in [Pou10] and [Kim07].

Application of an optical flow algorithm supposedly yields the highest quality depth map estimation using camera motion. But the results of the algorithm at that stage may be not good enough for several reasons. First, it is impossible to accurately determine optical flow for two frames in regions of opening and occlusion that appears when objects are in motion. In such regions depth can not be estimated correctly. Second, it is impossible to detect true motion in smooth areas, particularly in case of noisy video. Third, considering the high computational complexity of this stage, we must often sacrifice optical flow quality to increase processing speed; this affects final results. So, some postfiltering is required to reduce errors in a depth map or to make them less visible in the final result.

Such postfiltering can be performed using some variations of simple Gaussian smoothing [Zha05] or using more-complex filtering: for example, bilateral filtering [Cha09]. To address the problem of stereo correspondence this approach can be extended to use multilateral filtering, particularly with a left-right consistency metric, which makes it more robust. Details of this approach are presented in [Mue10] and [Jac10]. Other approaches under active development reduce the problem of depth estimation to a matter of energy optimization for the whole frame, a process that requires extensive processing time but produces better quality for the final results [Zha08].

2 PROPOSED METHOD

Our work involves spatial filtering of a depth map that was estimated using camera motion for single-view video. Our approch can be applied to depth maps generated by any other method, however, because the filtering does not use any additional information from the scene.

To suppress artifacts, we propose trilateral filtering. The convolution kernel is built for every pixel and is composed of the following components: Gaussian kernel G with a specified radius, matrix $I(x,y)$ based on the photometric difference between the current pixel with coordinates (x,y) and neighboring pixels in the source image, and analogous matrix $D(x,y)$ calculated for this pixel using a depth map.

G responds to the distance from the current pixel being processed: the farther the pixel is from the center the lesser influence it has on the result. Weights $i(x,y)$ in image-based component are linearly dependent on the difference between the central pixel and other pixels:

$$i(m,n) = \\ = \begin{cases} \frac{th_{color} - IDiff_{xy}(m,n)}{th_{color}}, & IDiff_{xy}(m,n) \leq th_{color} \\ 0, & IDiff_{xy}(m,n) > th_{color} \end{cases} \tag{1}$$

where $IDiff_{xy}(m,n) = (|red(m,n) - red(x,y)| + |green(m,n) - green(x,y)| + |blue(m,n) - blue(x,y)|)/3$, $m \in [x-r, x+r], n \in [y-r, y+r]$.

Depending on the input data, the color difference may be calculated in another way: for example, as the maximum absolute difference for the color components or as the absolute difference between the average values of the color components. But the results and processing speed vary just slightly so we selected the mean absolute difference as the more general approach. The parameter th_{color} is set accordingly to the source image's noise level and contrast range.

For the depth-based component $D(x,y)$, linear dependence is not applicable. In some models depth has only a few grades, so an error in one depth grade may yield too large a color range. Also, we chose to penalize large differences in depth, so we used a logistic function, and we calculated weights for depth-based component in the following way:

$$d(m,n) = 1 - \frac{1}{1 + e^{-t \cdot DDiff_{xy}(m,n) + 6}} \tag{2}$$

where $DDiff_{xy}(m,n) = |D(m,n) - D(x,y)|$, $m \in [x-r, x+r], n \in [y-r, y+r]$ and t is a parameter that influences on the acceptable deviation of the depth value from value in the central pixel. The constant 6 is based on the properties of the logistic function: the value of the function for arguments greater than 6 is very close to zero.

When we take into account information from a rough depth map, a problem may crop up. All the artifacts in the depth map will influence the depth component in the convolution kernel, and consequently, they will influence the final result. So, to calculate weights in the depth component, we need a depth map that is largely free of artifacts but that is not blurred, having strong edges. To solve this problem we used bilateral filtering with an adaptive threshold. If enough pixels of the same color are near the current pixel, we set the threshold low to preclude using pixel from another depth level and blurring of an edge. But when there are few similar pixels, we set the filtering strength high enough to suppress the artifacts. We choose the filtering radius according to the size of the artifacts that we want to suppress.

Then final convolution kernel $K(x,y)$ is calculated as the element-wise product of matrices G, $I(x,y)$ and $D(x,y)$.

$$k(m,n) = g(m,n) \cdot i(m,n) \cdot d(m,n) \tag{3}$$

The resulting pixel value in the filtered depth map is:

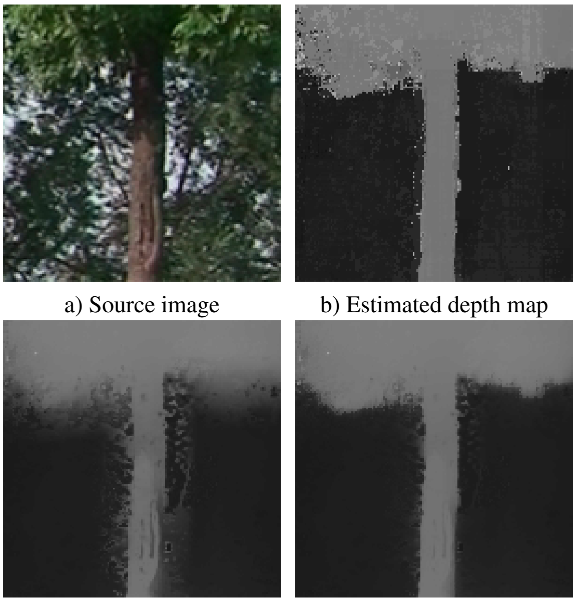
$$r(x,y) = \frac{\sum\limits_{m=x-r}^{x+r} \sum\limits_{n=y-r}^{y+r} k(m,n) \cdot z(m,n)}{\sum\limits_{m=x-r}^{x+r} \sum\limits_{n=y-r}^{y+r} k(m,n)}, \tag{4}$$

where $z(m,n)$ is the pixel in the estimated depth map.

Compared with bilateral filtering, the trilateral approach has some advantages. If we ignore information from the source image, we are only blurring a depth map and are not really enhancing it. But if we ignore depth when building a convolution kernel, we may blur a boundary between two objects of the same color. Also, when we use only color component we may obtain the wrong thin depth flows on boundaries, since boundary colors are usually the average of the objects they divide. An example of flows artifact is presented in Figure 1.

3 RESULTS

Figures 2, 3 and 4 show the results for the proposed method as well as a comparison with the bilateral approach. This method outperforms image-based bilateral algorithm in preserving the boundaries of objects detected by optical flow. Also, it produces smoother depth in uniform areas compared with the depth-based bilateral approach.



a) Source image

b) Estimated depth map

c) Result of bilateral filtering

d) Result of trilateral filtering

Figure 1: Example of flow artifact for the bilateral filter, sequence “Road”, frame 29

4 FUTURE WORK

In the short term, the authors plan to better integrate data from optical flow algorithm into postfiltering algorithm; this integration which will improve the restoration quality for small details and will also allow as to obtain a confidence measure for each pixel. Using this measure, we will be able to estimate the probability of that artifacts will appear in certain regions and allowing us to achieve better results.

Another direction in the algorithm’s development is use of temporal data from previous and subsequent frames compensated according to optical flow. This approach will significantly increase computational complexity but should improve the depth map’s temporal stability and improve details in a frame.

Also we plan to use the source image and optical flow data to extract separate objects as structural units for more precise processing of object boundaries.

5 CONCLUSIONS

In this paper, we proposed an algorithm of trilateral postfiltering for depth maps created from camera motion. We compared this algorithm with other approaches, and we described and demonstrated the relative advantages of our approach. After reviewing potential problems that can appear when using an inaccurate depth map for calculating the convolution kernel, we described our method of solving these problems. Lastly, we described our intended directions of future work.

ACKNOWLEDGMENTS

This research was partially supported by grant 10-01-00697-a from the Russian Foundation for Basic Research.

REFERENCES

[Sax06] Ashutosh Saxena, Sung H. Chung, *Learning Depth from Single Monocular Images*. Advances in Neural Information Processing Systems, 2006.

[Zhu09] Shaojie Zhuo, Terence Sim, *On the Recovery of Depth from a Single Defocused Image*. Proceedings of International Conference on Computer Analysis of Images and Patterns (CAIP), vol. 5702/2009, p. 889-897, 2009.

[Bat04] S. Battiatoa, S. Curtib, M. La Casciac, M. Tortorac, *Depth-Map Generation by Image Classification*. Proceedings of SPIE, vol. 5302, 95, 2004.

[Pou10] Mahsa T. Pourazad, Panos Nasiopoulos, Rabab K.Ward, *Generating the DepthMap from the Motion Information of H.264-Encoded 2D Video Sequence*. EURASIP Journal on Image and Video Processing, Volume 2010.

[Kim07] Donghyun Kim, Dongbo Min, Kwanghoon Sohn, *Stereoscopic Video Generation Method Using Motion Analysis*. Proc. of the 3DTV Conference, p. 1-4, 2007.

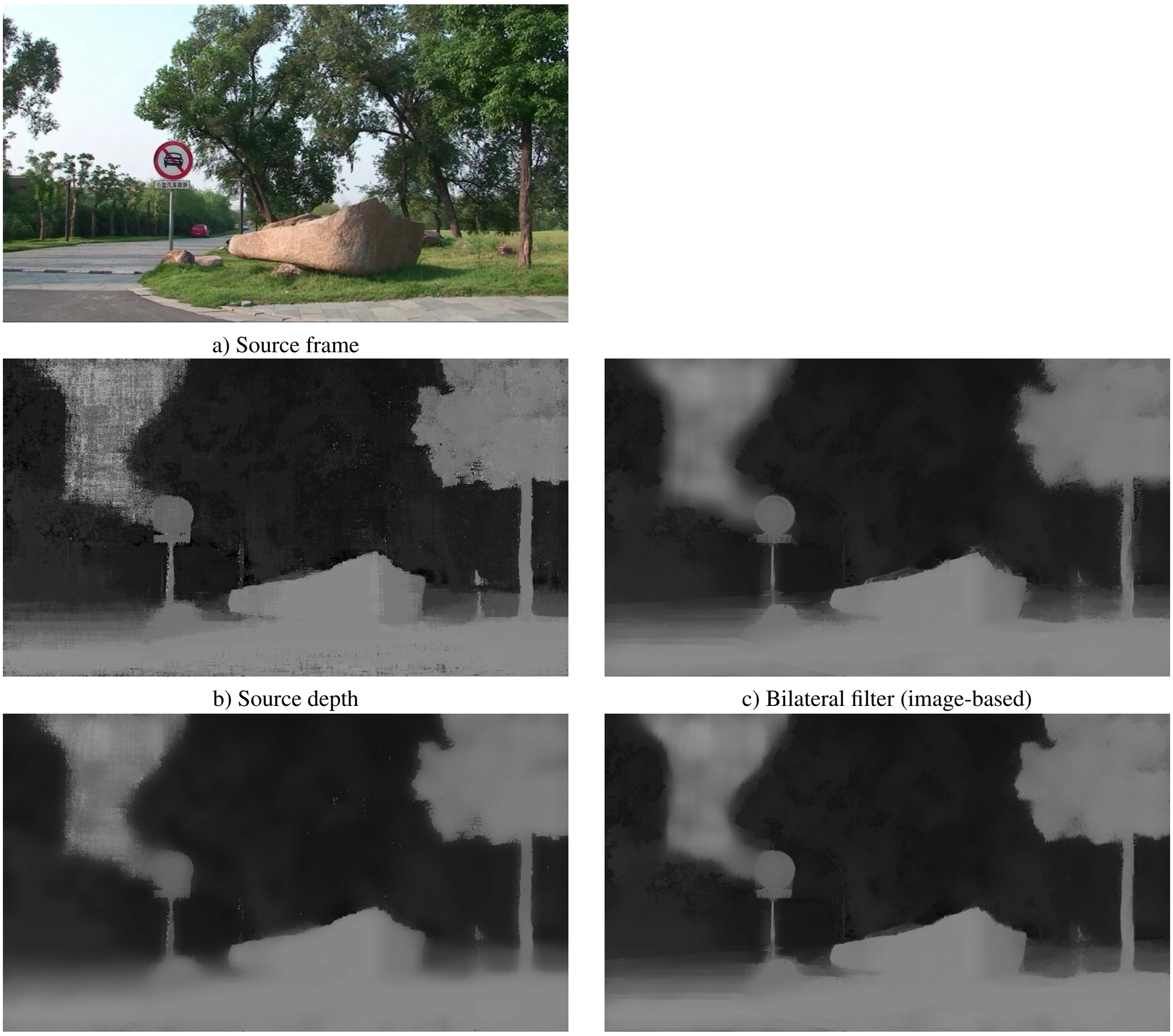
[Zha05] Zhang, L., Tam, W. J., *Stereoscopic Image Generation Based on Depth Images for 3D TV*. IEEE Trans. on Broadcasting, vol. 51, pp. 191-199, Jun. 2005.

[Cha09] Chao-Chung Cheng, Chung-Te Li, Po-Sen Huang, Tsung-Kai Lin, Yi-Min Tsai, and Liang-Gee Chen, *A Block-based 2D-to-3D Conversion System with Bilateral Filter*. International Conference on Consumer Electronics, p. 1-2, 2009.

[Zha08] Zhang, G., Jia, J., Wong, T., Bao, H., *Recovering Consistent Video Depth Maps via Bundle Optimization*. Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, p. 1-8, 2008.

[Mue10] Mueller, M., Zilly, F., Kauff, P., *Adaptive cross-trilateral depth map filtering*. 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), p. 1-4, 2010.

[Jac10] Jachalsky, J. Schlosser, M. Gandolph, D., *Reliability-aware cross multilateral filtering for robust disparity map refinement*. 3DTV-Conference: The True Vision - Capture, Trans-



a) Source frame

b) Source depth

c) Bilateral filter (image-based)

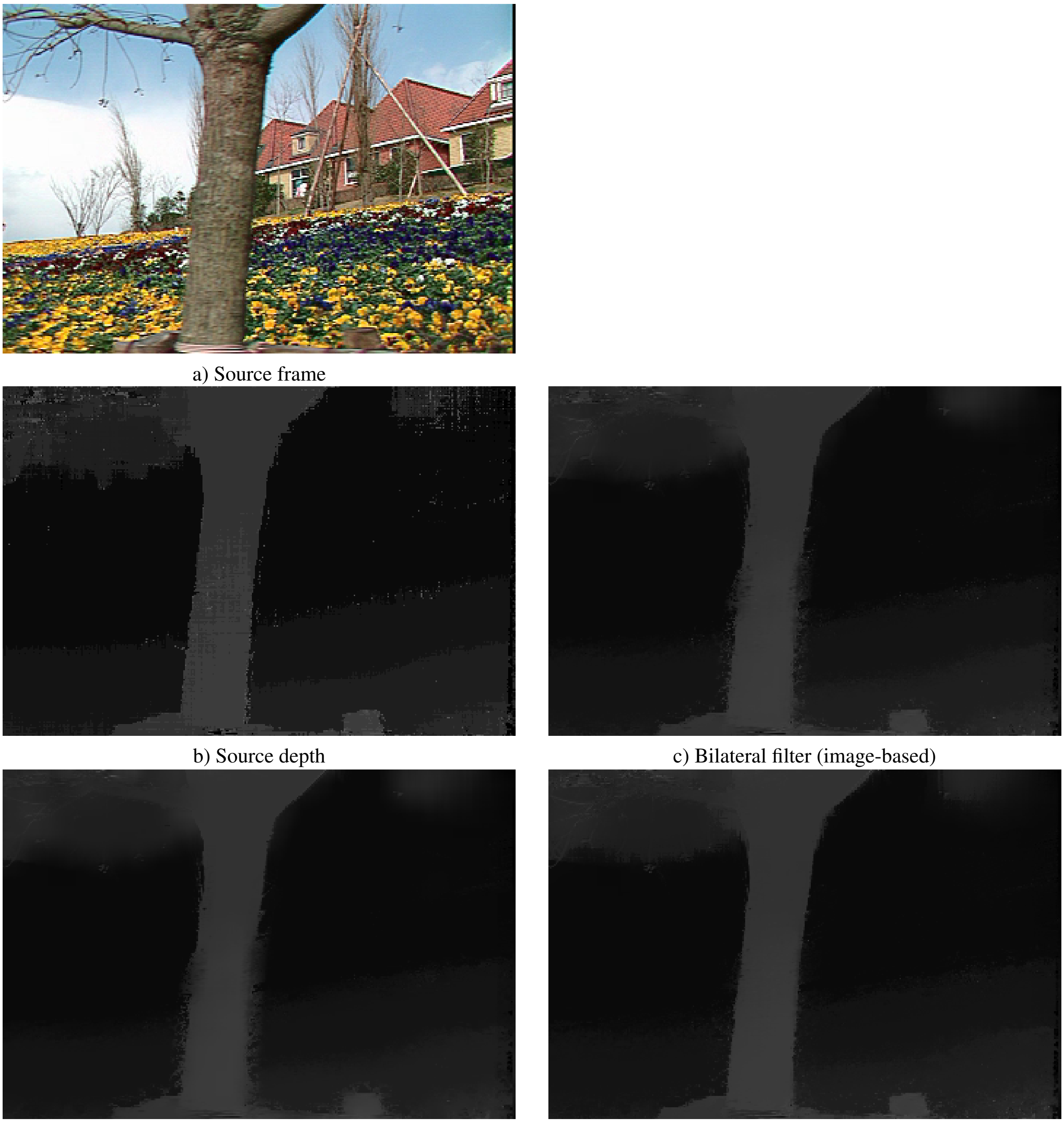
d) Bilateral filter (depth-based)

e) Trilateral filter

Figure 2: Comparison of different filtering methods for “Road” sequence, frame 29.

mission and Display of 3D Video (3DTV-CON), p. 1-4, 2010.

[Jun10] Jae-Il Jung, Yo-Sung Ho, *Depth map estimation from single-view image using object classification based on Bayesian learning*. 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), p. 1-4, 2010.



a) Source frame

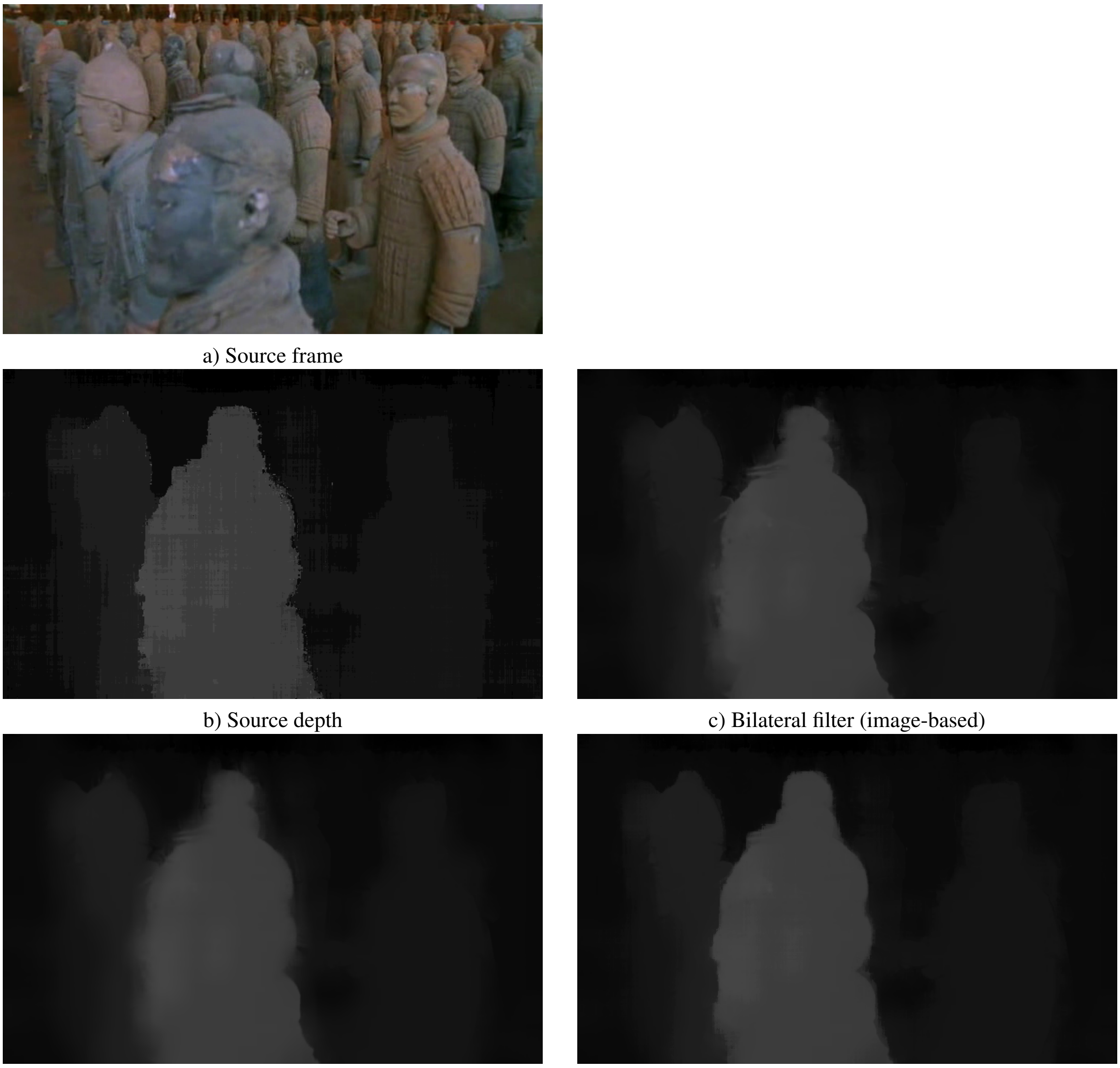
b) Source depth

c) Bilateral filter (image-based)

d) Bilateral filter (depth-based)

e) Trilateral filter

Figure 3: Comparison of different filtering methods for “Garden” sequence, frame 18.



a) Source frame

b) Source depth

c) Bilateral filter (image-based)

d) Bilateral filter (depth-based)

e) Trilateral filter

Figure 4: Comparison of different filtering methods for “Warrior” sequence, frame 17.